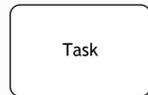


## Attività



Un **task** è un'unità di lavoro, cioè il lavoro da svolgere. Quando si annota con il simbolo indica un sottoprocesso, cioè un'attività che può essere perfezionata.



Una **transazione** è un insieme di attività che si legano logicamente; essa potrebbe seguire uno specifico protocollo.



Un **sottoprocesso basato su eventi** si trova all'interno di un processo o sottoprocesso. Si avvia quando il suo evento di inizio viene attivato e può interrompere il processo di livello superiore oppure eseguire in parallelo (senza interruzioni) in base all'evento di inizio.



Una **call activity** è un contenitore di un sottoprocesso definito globalmente o un task che può essere riutilizzato nel processo attuale.

### Simboli per attività

I seguenti simboli indicano il comportamento di esecuzione delle attività:

- Sottoprocesso
- Loop
- Esecuzione in parallelo
- Esecuzione sequenziale
- Ad hoc
- Compensazione

### Tipologie di tasks

Le tipologie specificano la natura dell'azione da eseguire

- Task di invio
- Task di ricezione
- Utente
- Task manuale
- Regole di business
- Service
- Script

### Flusso sequenziale

definisce l'ordine di esecuzione delle attività.

### Flusso predefinito

è il ramo predefinito da scegliere se tutte le altre condizioni vengono valutate come false.

### Flusso condizionale

ha una condizione assegnata che definisce se usare o meno il flusso.

## Conversazioni



Una **comunicazione** definisce un insieme di scambi di messaggi collegati logicamente. Se annotati con un simbolo indicano una comunicazione interna ad un'altra conversazione.

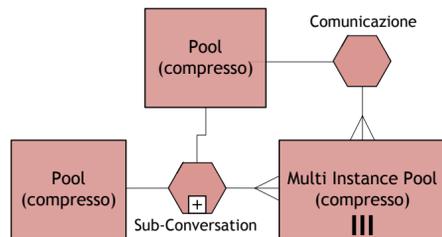


Un **conversation link** connette le comunicazioni ed i partecipanti.

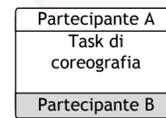


Un **forked conversation link** connette le comunicazioni e molteplici partecipanti.

### Diagramma di conversazione



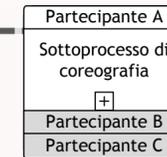
## Coreografie



Un **Task di coreografia** rappresenta un'interazione (scambio di messaggi) tra due partecipanti.

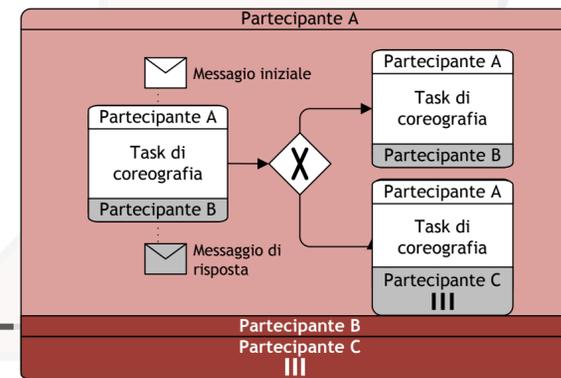


Il simbolo **Multiple Participants** denota un insieme di partecipanti della stessa tipologia.

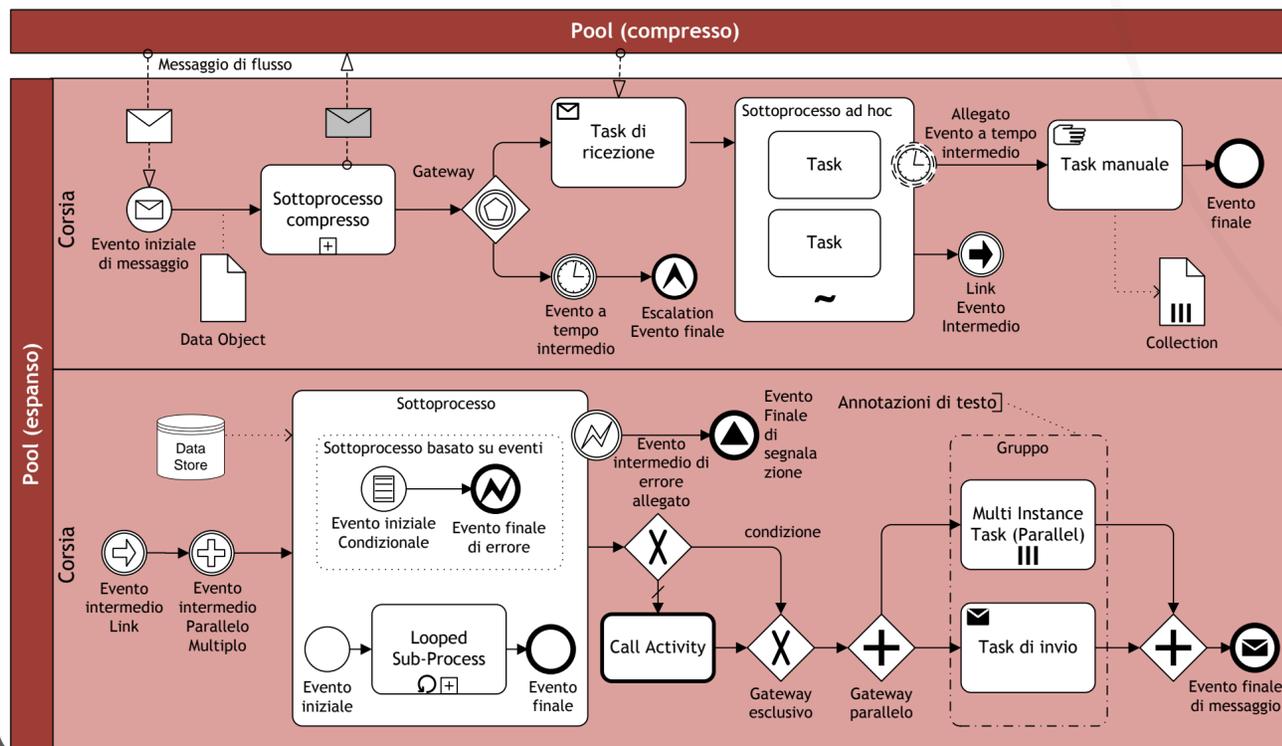


Un **Processo di coreografia** contiene una coreografia rifinita con molte interazioni.

### Diagramma di coreografia



## Collaboration Diagram



## Eventi

	Alto livello	Inizio	Intermedio	Fine		
	Interruzione di sottoprocessi	Non-interruzione di sottoprocessi	Catching	Boundary Interrupting	Boundary Non-Interrupting	Throwing
<b>Non definiti:</b> punti di inizio, cambi di stato, o stati finali.						
<b>Messaggio:</b> invio e ricezione di messaggi						
<b>Timer:</b> eventi a tempo.						
<b>Escalation:</b> passa ad un livello più alto di responsabilità.						
<b>Condizionale:</b> reagisce a condizioni di business cambiate o integra regole di business.						
<b>Link:</b> Due corrispondenti link events sono uguali ad un flusso sequenziale.						
<b>Errore:</b> attiva o si occupa di un errore.						
<b>Cancel:</b> reagisce a delle transazioni cancellate o causa una cancellazione.						
<b>Compensazione:</b> gestisce o innesca la compensazione.						
<b>Signal:</b> comunica con più processi. Lo stesso segnale può essere intercettato più volte.						
<b>Multiplo:</b> intercetta uno tra vari eventi. Gestisce tutti gli eventi definiti.						
<b>Parallelo Multiplo:</b> intercetta tutti gli eventi.						
<b>Terminate:</b> causa la fine immediata di un processo.						

## Gateways

### Esclusivo (xor)



In caso di **splitting**, il flusso sequenziale viene diretto esattamente verso uno dei rami in uscita. In caso di **merging**, il flusso aspetta che un ramo in entrata arrivi a termine prima di andare avanti.

### Basato su eventi



Questo simbolo è sempre seguito da intercettazioni di eventi o **tasks** di ricezione. Il flusso sequenziale prosegue verso il successivo **task**/evento che accade per primo.

### Parallelo



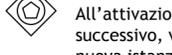
Quando viene usato per dividere il flusso sequenziale, tutti i rami in uscita sono attivati simultaneamente. Invece quando viene usato per unire rami paralleli, il flusso aspetta il completamento di tutti i rami in entrata prima di andare avanti.

### Inclusivo



In caso di **splitting**, uno o più rami sono attivati. Il flusso va avanti solamente quando l'esecuzione di tutti i rami è terminata.

### Esclusivo basato su eventi



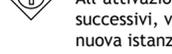
All'attivazione di ogni evento successivo, viene avviata una nuova istanza di processo.

### Complesso



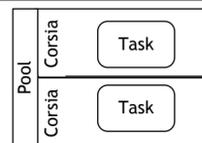
Gestioni di **merging** e **branching** che non sono gestite da altri gateways.

### Parallelo basato su eventi



All'attivazione di tutti gli eventi successivi, viene avviata una nuova istanza di processo.

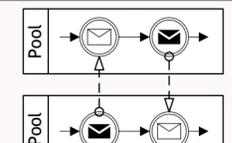
## Swimlanes



**Pools (Partecipanti) e Lanes (corsie)** rappresentano le responsabilità per le attività in un processo. Esse possono essere un'organizzazione, un ruolo o un sistema. Le corsie suddividono le **pools** o altre **corsie** gerarchicamente.



**Flusso di messaggi** rappresenta il flusso di informazioni. Un flusso di messaggi può essere unito a **pools**, attività, o eventi di messaggi.



L'ordine degli scambi di messaggi può essere specificato associando il flusso di messaggi e il flusso sequenziale.

## Data



Un **Data Input** è un input esterno usato all'interno del processo. Può essere letto da un'attività.

Un **Data Output** è una variabile disponibile come risultato di un intero processo.



Un **Data Object** rappresenta le informazioni che attraversano l'intero processo, come ad esempio documenti di business, e-mails, o lettere.



Un **Collection Data Object** rappresenta una collezione di informazioni, come ad esempio una lista di elementi ordinati.



Un **Data Store** è un luogo dove il processo può leggere oppure scrivere dati, ad esempio un database. Esso si mantiene oltre la durata dell'istanza del processo.



Un **messaggio** è usato per rappresentare i contenuti di una comunicazione tra due partecipanti.